CATS NEWSLETTER

Presidential Ramblings

Tim's enthusiasm for his new role is close to amazing. Knowing of his June marriage in California I certainly expected him to have second thoughts about producing a NL before the July meeting. Instead, he's putting one together before the big event so he can meet his own deadline. Tim, you're the next best thing to sliced bread!!

The only difficulty Tim's early bird schedule posed for me is that I'm writing this both before I'm elected and before I even leave for Milwaukee. If I'm not reelected, the Curnutts will have to call to "Stop the Presses" of the printer. But, assuming I do take office, I want to share with you the fact that my main goal will be to involve many different voices in active roles this year. I heartily second Joe's eloquent plea for volunteerism expressed in his May presentation. Furthermore, I'm convinced that many members have talents that could and should add new dimensions to the CATS organization.

Goal accomplished, this time next year I expect my role to be "Immedeate Past President"!!!

Tom Robbins is accepting the baton from George Rey, becoming the one who plans the monthly Program of the Day. I don't think he'll object to my asking you to not only give him suggestions, but to offer to present a program yourself or assume responsibility to obtain a speaker.

By now you should have cottoned to the fact that my theme is "What are YOU going to do for CATS?" Next month I'll announce my latest appointments so you can see the great company you'll be joining when YOU volunteer.

In closing I'm sure I'm safe in saying that you all join me in wishing Catherine and Tim Swenson every happiness in their life together.

From The Editor:

As Ruth mentioned in her column, I am getting this issue completed before I leave for California and before the June meeting. I would much rather have the newsletter done early than have it be late.

After this issue I have a few articles for the next issue, but I can always use more. The more articles I have, the thicker the newsletter will be. I have a few ideas for articles myself, but I don't want to have my name on every second article. So, hit those word processors, folks.

For those interested in older computers (that includes the Sinclair line), a Ham Radio Fest is the place to be. I just returned from the Manassas Ham Fest with a few extra items to add to my collection. Don't expect to find what you are looking for, but expect to find odd things at good

Contents

July Meeting Topic

"A Look At Artificial Intelligence"

Cats will have a representative of one of the leading professional societies in the Washington Metropolitan area give a presentation on what AI is, how it grew, and how it may be working for us even more in the near and long-term future. This will be a great opportunity to become familiar with one of the great hardware/software breakthroughs of the 80's!!

prices. Some people are just giving equipment away to those interested in it (this was how I got a PDP-11/05 minicomputer). There is an occasional Sinclair product. Don't limit yourself to just Sinclair computers. There is a wide variety of old, used, CHEAP, computers out there. For me, they are fun to play with. They each have their own personality. So, if you ever get a chance to go to a Ham Fest, go. It's worth a look.

That's about all I have to say. Hope you enjoy the newsletter and I'll see you at the July meeting.

The Z88 and Telecommunications
By Dave Bennett

The Z88 is particularly well suited for communicating to modems and other computers. A serial port which communicates up to 38,400 baud, a VT52 terminal program and a dedicated file transfer utility are included with the stock Z88. The Z88 also weighs only 2 pounds and is easily transported right next to the device you want to communicate with.

My Z88 is mainly used to talk to a pocket modem and a QL. Z88 files are archived on a QL disk. I also occasionally communicate with PCs at work and other Z88s.

I have designed and built cables to communicate with the QL, PCs and other Z88s. I built the modem cable from recommended pinouts by Cambridge.

The QL cable is as follows:

```
Z88                QL SER2
---                -- ----
```

```
  2 ---------------- 3
  3 ---------------- 2
!-4              ----- 4
!-5            !     5
  6            !     6
  7 ----------!---- 7
!-8 ----------!     8
!-9                 9
```

I have tried several other designs for a Z88 - QL cable but have not found a cable that works as well for transfering files.

What I use to transfer files are Q_Link for the QL and Z88COM for the Z88. Q_Link is a shareware terminal program from Harvey Taylor. It includes CRC and Checksum xmodem as well as many other features. Z88COM is a publicly available terminal program from Phil Wheeler. It extends the built-in VT52 with checksum xmodem and some other features. Phil also operates the Z88 BBS in Los Angeles.

I basically connect the computers with the cable and run terminal programs which utilize xmodem on each machine. You set up one machine to send and one to receive. Then you start the transfer on both machines at the same time.

You can transfer ascii files without xmodem by various means but I prefer to use xmodem. It provides error checking in the transfer so you know the files are transfered OK. You can also transfer any type of file even machine language.

Z88 to PC and Z88 to Z88 transfers work similarly. On the PC I prefer to use PROCOMM. It is a powerful program and extremely easy to use. I would like to see it made available for the QL. For Z88 to Z88 transfers I am using Z88COM on both machines.

The Z88 - PC and Z88 - Z88 cables are as follows:

```
Z88 DB9         PC DB25
--- ---         -- ----

  2 ------------- 3
  3 ------------- 2
!-4              4-!
!-5              5-!

  6              6-!
  7 ------------- 7 !
  8 -------!      8-!
           !
           !----- 20
```

```
Z88             Z88
---             ---

  2 ----------- 3
  3 ----------- 2
!-4             4-!
!-5             5-!
  6             6
  7 ----------- 7
  8 ----------- 9
  9 ----------- 8
```

There also is a variation on Z88COM called Link by Dave Armstrong which turns the Z88 into a mini BBS. There are no message facilities but you can transfer files. All typing is done at a remote computer. The Z88 just sits there. It is similar to PROCOMMs Host Mode. It works well connected to a computer in the same room. Modem facilities were removed from this variation but you can connect to the modem and then switch to Link by hand. With this program you can transfer files with someone across town at a prearranged time. You could even transfer QL and 2068 files.

There also is another program available for the Z88 called ZFU by Richard Haw. This is a utility which takes many files and archives them into one large file. In the process it compresses the files so the resulting large file is much smaller. You can archive all the files in all the ram devices of the Z88 into one file and just backup that one file to your QL or whatever.

The Z88 also works nicely with any Hayes compatible modem. Here are the official modem pinouts from Cambridge:

```
Z88            Modem
---            -----

2 ----------- 2
3 ----------- 3
7 ----------- 7
5 ----------- 20
8 ----------- 20
9 ----------- 20
```

The Z88 includes a built-in VT52 terminal emulator. It works quite nicely although you have to dial manually with Hayes AT commands. It is also easy to send and receive ascii files with just the built-in software. The built-in VT52 also connects quite nicely to a VAX mini computer once you tell the VAX you have a VT52 with 8 lines.

But I like Xmodem. A terminal program was developed in England which provides xmodem. The disadvantages of this program are its high cost and it also adds its own protocol in place of the VT52. The built-in VT52 works much better.

Fortunately Z88COM came out which provides xmodem but works in concert with the built-in VT52. Z88COM is also much cheaper. In most cases it is available for the cost of a download. Z88COM can also autodial and autologon to five systems. It has a message upload feature for sending text files. All in all it is a great program.

The latest I heard was that Cambridge just came out with a VT100 terminal program. I have not heard too much about this. The VT100 can display graphics from a remote IBM or VAX. It moves the cursor around your screen on command from the remote system. I am not too sure how well this will work on the Z88 with its 8 line screen but it sounds interesting.

The Z88s portability and sheer connectivity to anything speaking RS232 make it a great little computer for telecommunications.

The C Column
By Tim Swenson

As promised last month, I'll talk about how C accesses files.

The simplest way to discuss how C handles files is to take a look at a C program that uses it. Look at program number 5. This program asks the user for a file name. It then opens the file, reads in a character and prints it out the screen. This goes on for the entire file.

The program declares a character array of size 20 to be hold the file name. It then declares the integer fd to hold the file ID. When a file is opened, C assigns a number to it. This number is returned from the fopen command to the variable fd.

Note the syntax of the fopen command. The file name is sent into the function along with a string that specifies if the file is for reading or writing. If we wanted to open a file to write to, we would have used "w" instead of "r".

The next couple of lines checks to see if the file was opened sucessfully. If the file was not opened (there was some error like a bad file name) fd will be returned as NULL (a predefined constant).

Instead of using getch the function getc is used. Where getch defaults to the keyboard, getc needs a file ID to know where to get it's input. In our case we use fd.

We exit the while loop when the program reaches the end of the file (EOF). After the program exits the while loop, we use the fclose function to close the file. We must tell fclose which file to close. If we do not close the file, we may loose the file.

```c
/* prog 5 */
#include <stdio_h>

    main() {
        char    file[20];
        int     c, fd;

        printf("Enter file: ");
        gets(file);

        fd=fopen(file,"r");
        if (fd == NULL) {
            printf("File open Error");
            abort(1);
        }
        while ((c=getc(fd))!=EOF) {
            putchar(c);
        }
        fclose(fd);
    }
```

Dealing With The BBS
By Mark Fisher

GETTING ON THE AIR

What you'll need:

This will take your computer, a terminal program, and a MODEM. For the 2068, the most common setup is the 2050 (Anchor) MODEM, along with SmartTerm II. For the QL, the horizon is wider - any "external" MODEM will work at 1200 BAUD. For terminal programs, again, the choice is wider, with QL Term, QLink, QL-52 being the main contenders. (If you want 300 BAUD compatibility, you'll have to do more work; i.e. the Modaptor and QCode software)

Now you'll have to connect the pieces:

2068 --- Plug the 2050 into the expansion bus in the back of the computer, and plug its adapter into the wall.

QL --- Get a connecting cable to bridge the gap between the DB9 connector on the QL and th e DB25 on the MODEM.

Using the software:

Load the software. If you have the SmartTerm II cartridge, that's already happened by the time you see the first screen. QL programs are nearly that simple.

Terminal programs must make a distinction between input that controls the operation of the program, and input that is intended to be sent out the port of your computer. The usual way to describe these choices is the term "mode". "Command mode" controls the operation of the terminal program itself, while "terminal mode" sends your input out the serial port. When you first get on, make sure your terminal program is saving everything to its buffer. This will let you review what happened after you disconnect. You may want to save the (I)nformation files as permanent references.

Depending on your particular setup, you may dial the BBS via your phone, or through the auto-dial option of your terminal program. Many external MODEM's can be dialed by typing ATDP5880579 from terminal mode.

When you connect with the BBS, you will see a prompt to set the date (this is an addition I've made to get around the lack of a clock on our machine). If you get something wrong, you'll be prompted again. Don't

worry; the computer has plenty of time, and you can go around as much as you like. Next, you will be asked for your name. If I've guessed the spelling of your name correctly, you'll be asked for the password. The default password is VERNON, but you should change that as soon as possible. (To encourage you to do so, I've also set your message reading width to 20. You can change both preset values from the (U)pdate record menu choice.)

EXPLORING

Once you get past this, you'll be IN the CATS BBS. There are no time limits here. The BBS will not penalize you for making a mistake. The worst thing that can happen is that you'll have to 'unplug' your computer. The BBS will detect this, and re-set itself for the next caller.

If you think of the BBS as a house, there are three main wings. They are the Main menu, the Mail menu, and t he files system. Entry into each is heralded by a specific phrase. You can think of each phrase as being carved over the entrance to the wing (one of the weaknesses of this metaphor is that the actual location of the phrase will be in the bottom left corner of your screen, not top center).

The Main Menu Wing:

"Over the Door", you'll see:

PBBS Menu, Commands are ....

This is the general purpose section of the BBS. Here, you'll find out about general (I)nformation, recent uploads, other users, and your personal custom-

ization values. You can also do some basic chores, such as toggling the full display of the menu or the [more] prompt, (Y)akking with me, (J)umping to CPM, switching to the (M)ail menu, or saying (G)oodbye.

The Mail Menu Wing:

Over the door of this wing is :

Folder: General
MAIL Menu: Commands are ....

This wing offers ways to delve through the message base. The most basic are (E)nter a message and (R)ead one. In addition, you can scan the headers, either for a given subject, or a list of all subjects. As the message base grows, it will get harder to scan everything. To deal with this, the message base is divided into 9 FOLDERS. By switching between the folders, you can view all "FOR SALE" messages, all 2068 messages, etc.

There are sub-menus that appear as you go through the more complex operatons: for (E)nter, there is a menu that appears when you press <ENTER> twice. It offers you the option to (L)ist, (E)dit, or C)ontinue entering the message, or (S)ave the message as public or (P)rivate.

The CPM Attic:

Over the door is:

B3>

The last wing of the "PBBS house" is the files area. The rules here are different. You can think of it as the "attic" of the system, storing all those things you MIGHT use. There are no menus here - instead there

are a number of discrete commands you can invoke. CPM interprets the word following the B3> prompt as the file name of a machine code program. There is room after the name for further data – see the examples below. When you are adding that extra data, separate it from the command with ONE SPACE. The most basic are:

"BYE" Hangs up the phone and resets the BBS f or the next caller.

"PBBS" Returns to the Bulletin board program, so you can leave a message about what you found up in the CP M attic.

"D" This is the Directory command – producing an alpha sorted list of files.

"USER xx" This changes the user area (C PM's sub-directory method)

"TYPE-DN (your filename here)" This will type to the screen the contents of the file you specify. If you have cleared your capture buffer and opened it, you will end up with a copy of the file you requested. If you use TYPE-DN on COM or DOC files you will see some interesting screen art, and possibly hang up your computer, but it won't hurt the BBS.

"TYPE-UP (your filename here)" This program will create a new file on the BBS computer, and store whatever you send to it there. You can use your terminal program's ASCII file transfer routines to send your ASCII files to the BBS. It will be closed when you send a control-Z, and after a few seconds, you'll see the B3> again.

"XMODEM (your filename here)" The is the big gun of the files area. This program can work with the Xmodem part of your terminal program to transport ANY file to or from the BBS. Typing XMODEM alone will give you a short list of your options. To use it, type "XMODEM (your filename here)" at the B3> prompt. You will then enter command mode of your terminal program, and start up its xmodem file transfer procedure. The two programs will work together to shift the file between the machines. XMODEM knows that you will take a while to get your end going – there's a delay built in. If you take too long it will abort the transfer, but you can always try again. The transfer may also be aborted if there are too many transmission errors – often a second try will work fine.

"DEMO CPM" To see several of these commands in action, type DEMO CPM. There is an automated demo of the basics (I've put in a place for your input – don't just sit back!).

T R Y   U S   O U T !

The Game of Chaos
By Tim Swenson

NOVA is a science program that runs on PBS. On a past show they discussed the idea of chaos. At the begining of the show they showed a game called THE GAME OF CHAOS.

Here is how the game works. On a board are three points in a triagular formation, numbered 1, 2 and 3. You randomly place a point some where on the board (P1). You roll a three-sided die to get a number from 1 to 3. This will tell

you what point to refer to. In our example we rolled a 3. Now draw a point that is half way from your original point to the numbered point (P2). Now keep doing this many many times.

One would expect to get a random pattern to come from this game, or chaos. Well, one does get chaos from this game, but a structured chaos. Over many interations of this game a fractal pattern comes out.

As soon as I saw this on the program I know I had to give it a try myself. I did not even wait until the program was finished to start work on the QL program. Below is a listing of a program to do this. Type it in, run it, and wait a while. It takes some minutes for the pattern to come out. But the wait is well worth it. It is very interesting to see a fractal pattern come out of so simple an algorithm. Especially one that seems so random.

The program can easily be converted to the 2068 or the 1000 with little modifications. It may not look great on the 1000, but something can be worked.

```
100 CLS
110 SCALE 100,0,0
120 INK 0
130 POINT 10,10
140 POINT 50,90
150 POINT 90,10
160 LET x=RND(0 TO 99)
170 LET y=RND(0 TO 99)
180 INK 7
190 REPeat loop
200     LET new_x=x
210     LET new_y=y
220     LET roll=RND(1 TO 3)
230     IF roll=1 THEN
240         big_x=10
250         big_y=10
260     END IF
270     IF roll=2 THEN
280         big_x=50
290         big_y=90
300     END IF
310     IF roll=3 THEN
320         big_x=90
330         big_y=10
340     END IF
350     LET x=(new_x + big_x)/2
360     LET y=(new_y + big_y)/2
370     POINT x,y
380 END REPeat loop
```

QL Small-C Companion
By Tim Swenson

I have been using the Small-C compiler on the QL for some time now. I have found the documentation severly lacking. Knowing that other users are probably having the same problem, I decided to document the solutions to some of the problems I ran across. I also looked into other versions of Small-C for other computers. A copy of this document is available from the QL library. The library has a hard copy of the document and a Quill file. The hard copy was printed out using a laser printer and looks good. I've utilized multiple fonts to enchance it's looks.

When looking at the CP/M version of Small-C, I found that the QL version did not come with all of the functions found in most Small-C compilers. I typed in the functions that I could convert and created a new library. I tested the library generator with this new library. The library is also available from the QL library. The functions are documented in the Companion documentation.

I'm all for computer dating,
But I wouldn't want one to marry my sister.

Additions to the QL Library
By Tim Swenson

Over the years I have written a number of usefull programs for the QL. Some of them I have used in articles for numerous newsletters. I have donated them to the club's QL library. I thought I would let users know about the more interesting and usefull.

Lister - This program reads in a SuperBasic, Pascal, or C program (actually any text file will work). It prints the program to the printer with page breaks and the file name at the top of each page. It even has options for NLQ, bold, etc, type print. It is set up for an Epson compatable printer.

QLcryptII - This programs reads in any file and encrypts it. The user enters a password that controls how the file is encrypted and decrypted. The output file may look like garbage, but when decrypted the original file is returned. Even works on executable files. It is very secure.

Calc - This is a small calculator program that works in decimal, hexidecimal, and binary. Lets you add any of the three bases and convert between them. Easy way to convert between bases. If compiled, it can be a nice pop up type calculator.

Maillist - A menu driven Archive program for maintaining a mailing list. User can add, edit, delete, and print entries. Has a nice data entry screen.

All of the programs have a Quill file that describes the programs fully. All of the documentation is 1 to 5 pages long. Maillist has longer documentation. Be aware that the software has my old address in California. I've been too lazy to update them.

See Tom Robbins for getting copies of the QL Public Domain library.

More On Benchmarks
By Herb Schaaf

Duane Parker had the time down to 23 seconds using Forth in his January classic on benchmarking with Pascal, QL-SuperBasic, etc.

The Small-C QL combo does it in 16 seconds(14 with Lightning installed). An IBM PS/2-30 and the same (well, nearly the same) source code using Turbo-C took 6 seconds. Mix-C & IBM did it in 5. A Toshiba T5200/100 & Turbo-C took 2 seconds.

When ported over to the VAX-8650 and timed it came back as 0.0 seconds, so I extended the run down to 1; then it took 0.2 seconds. Times to go down to 1 are:

Turbo-C & T5200	22 secs.
Mix-C & IBM	42
Turbo-C & IBM	55
Small-C & QL	155
with QL &	
Conqueror Mix-C	1390
Turbo-C	2070

Now as to costs:

Hardware----
    QL systems $$ hundreds
    IBM systems $ thousands
    T5200 $ ten thousand
    VAX $$hundreds of Thousands

Software----
    Small-C $ 1 (cost of disk)
    Mix-C $ 20 (BEST BUY!!)
    Turbo-C $ 100 (discount)
    UNIX C $ a lot more

```c
/*   dpbench7_c      May 13 1990   10:30am
**  Small-C benchmark program by Duane H. Parker
**  Ref:  CATS Newsletter Jan 1990  pages 6-8.
*/

#include <stdio_h>      /* standard input/output header    */
#define MAX 32767              /*  limit of signed integers    */

main()
{
char c, string[20];     /* characters and/or strings       */
int start, stop, index, i, j; /* handy integer values       */
int testfactor, tfsq;       /* testfactor and its square    */
int secs, d1[2], d2[2];     /* used in timing               */
int minval, pos;        /* more integers                   */
minval = 0;
pos = (MAX/100);
testfactor = 181;          /* int(sqrt(MAX))                */
tfsq = 32761;
    while( ((minval>0) + (minval<MAX) + (index == 1)) != 3)
    {
    stop = 0;              /* initializing for clean start     */
    index = 1;
    start = 1;
    cls(2);
    printf("This is a Small-C program\n");
    printf("of a benchmark test by DH Parker\n");
    printf("see CATS Newsletter of January 1990\n");
    printf("\nStop calculation at what value less than %d ?\n",MAX);
        gets(string);         /* when to stop ? */
        j = strlen(string);
 strncpy(c,string,1);
        start = atoi(c);         /* makes string into integer  */
    for(i=0 ; i<j; i++)    /* checks for correct input    */
            {
            c = string[i];
      stop = isdigit(string[i]); /* is it a digit ?   */
            index = stop * index;
        }
    minval = atoi(string);
    if( j>5 )               /* are there too many digits ?  */
    {
    index = 0;
        }
    if((j == 5) && (index == 1) && (start>3) )
        {
                index = 0;
            }
    if((minval>=MAX) || (minval<1) || (index == 0))
            {
        printf("\n\nI N V A L I D  ! !"); /* error trapping */
        printf("\n\nany key to try again\n q to quit");
        c = getchar();
        switch (c)
        {
        case 'q' :  xcit();  /* 'fancy' exit    */
        }
    }
    ]
    cls(stdout);       /* clear screen */
    printf("Ready to find primes from 32767 to %d\n",minval);
    printf("any key to begin\n"); /* inform and wait for */
    getchar();                    /* trigger    */
    {
                    /* start clock    */
    *d1 = date();     /* starting time  */

    for(index = MAX; index >= minval ; index -= 2)
        {
 while ((pos*100)> index) /* make newline to    */
            {                       /* group by 100's     */
      --pos;
      printf("\n");
            }
        while( index < tfsq )    /* reduce tfsq and */
        {                        /* testfactor when */
```

```
                    for( j = 0; j < 4 ; j++)    /* possible  */
                       {
                       tfsq -= testfactor;
                       tfsq++;
                       }
                           testfactor -= 2;
                    }
                    if(testfactor == 1 && index != 1 )
                    {                            /* small primes */
                    printf("%6d",index);
                    }
                       for( j=3; ( j <= testfactor ); j += 2 )
                       {
                       if((index % j)==0) /* not prime      */
                          break;
                       if(j >= testfactor) /* is prime       */
                    printf("%6d",index);
                       }
                    }
         if(minval < 3)      /* to catch 2,     */
          {                   /* the even prime */
          printf("%6d",2);
                 }
         }
         *d2 = date();           /* checks time of finish   */
         secs = d2[1] -d1[1];    /* simple difference       */
         printf("\nRuntime = %d seconds in Small-C\n",secs);
         printf("\nany key to exit");
         getchar();
         xcit();
}
xcit()          /* a fancy (?) way to exit     */
{               /* a function is like a defined procedure */
    mode(8);
    cls(2);
    at(stdout,1,8);
    ink(stdout,1);
    printf("It's been good to");
    csize(stdout,3,1);
    at(stdout,3,11);
    ink(stdout,2);
    flash(stdout,1);
    printf("'C'");
    flash(stdout,0);
    csize(stdout,0,0);
    at(stdout,12,15);
    ink(stdout,4);
    printf("you");
    at (stdout,17,5);
    flash(stdout,1);
    ink(stdout,6);
    printf("any key to exit . . ._");
    getchar();
    mode(4);
    flash(stdout,0);
    exit(2);
}
/* The end of the small-c source code for dpbench7_c May 13, 1990 11am  */
```

The Capital Area Timex/Sinclair Users Group is a not-for-profit group devoted to serving the interests of those who own, use, or are interested in the Timex/ Sinclair family of computers.

Executive Board

President	Ruth Fegley
1st Vice President	Hank Dickson
2nd Vice President	George Rey
Recording Secretary	Audrey Curnutt
Corresponding Secretary	Joe Miller
Treasurer	Barry Washington
Members-at-Large	Mark Fisher
	Bob Curnutt
	Akin Olowofoyeku
Immediate Past President	Tom Bent
Newsletter Editor & contact person	Tim Swenson (703) 820-6657

CATS maintains a gratis exchange of newsletters with approximately 30 Users Groups across the U.S. and Canada. Groups not sending a newsletter to us for six months are automatically taken off the list.

Meetings

Monthly meetings are held from 11:00 AM to 4:30 PM, on the second Saturday of each month, at the New Carrollton Public Library.

Newsletter

Memberships cost $15 per year, are good for 12 months, and include all privileges (access to libraries, group buys, etc.). A newsletter only subscription is available for persons living outside the Washington Metro area and is $12 per year.
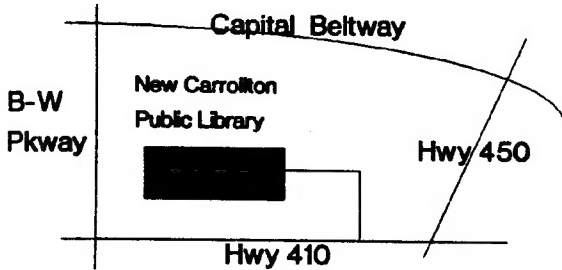
Networks

Timex SIG on Compuserve: Wednesday night, 10:00 PM Eastern time (GO CLUB).
QZX BBS: (505) 522-7081 FIDO net 15, node 18, node 9.
6. East Coast dial (703) 547-4815 FIDO net
CATS BBS: (301) 588-0579, 8-N-1, 300/1200 baud. News and general information, 24 hours a day.

B-W Pkway — Capital Beltway — New Carrollton Public Library — Hwy 450 — Hwy 410

The next meeting of CATS will be held on: July 14, 1990

11:00 AM Hardware Workshop
2:00 PM General Meeting

At: New Carrollton Public Library 7414 Riverdale Road (Hwy 410), New Carrollton, MD

IF YOU ARE NOT A MEMBER OF CATS. THIS IS THE ONLY ISSUE YOU WILL RECEIVE

PLEASE NOTE OUR NEW ADDRESS